

# An Adaptable WSN-based Flood Monitoring System

Danny Hughes<sup>1</sup>, Gordon Blair<sup>1</sup>, Geoff Coulson<sup>1</sup>, Phil Greenwood<sup>1</sup>, Barry Porter<sup>1</sup>,  
Paul Smith<sup>2</sup> and Keith Beven<sup>2</sup>.

<sup>1</sup> Computing, InfoLab 21, South Drive, Lancaster University, Lancaster. LA1 4WA  
{danny, gordon, geoff, greenwop, barry.porter}@comp.lancs.ac.uk

Environmental Science Department, Lancaster University, Lancaster.LA1 4YQ  
{p.j.smith, k.beven}@lancaster.ac.uk

**Abstract.** Flooding is a serious and increasing problem. As a result, there is a need to deploy more sophisticated sensor networks to detect and react to flooding. This paper outlines a demonstration which showcases a deployed flood monitoring platform, which uses WSN technologies and next generation middleware to improve support for flood warning.

**Keywords:** WSN, Grid, Environmental Monitoring, Middleware.

## 1 Introduction

Current flood monitoring approaches involve deploying depth and flow sensors in flood-prone areas, and feeding the collected data to grid-based computational models which predict flood events [1] [2]. We believe that there is considerable scope for improvement in such scenarios. In particular, we focus upon shifting the execution of prediction models to the wireless sensor network (WSN) itself, which acts as a ‘mini-grid’. Computations organised in this way are used to provide more timely warnings, and to optimize the WSN for predicted environmental conditions. For example, the network can employ a low power, low performance organisation in quiescent conditions, and switch to a high power, high performance organisation when flooding is imminent. This is supported by our ‘GridStix’ WSN/Grid platform.

The GridStix software platform is built on our reconfigurable component-based GridKit middleware [4]. This provides the key functionality required to develop both WSN and grid applications. GridKit is based on our language-independent OpenCOM component model [5] with each area of functionality implemented as an independent component framework. As it is component based, GridKit is inherently configurable and extensible, allowing us to build rich support for Grid and WSN applications, or, conversely, to build minimal deployments suitable for execution on embedded hardware [6]. Network support is provided by GridKit’s ‘Open Overlays’ framework, which supports ‘pluggable’ networks which can be instantiated dynamically.

The GridStix hardware is based on the Gumstix embedded computer [3], which is so named as it is roughly the size of a pack of gum. GumStix are more powerful than highly-embedded devices such as Berkeley Motes [6]. Each features a 400MHz XScale CPU, 64MB of RAM, 16MB of flash and Bluetooth. Additionally, each node is extended with 802.11b and low power 433MHz radios. A subset of nodes is also

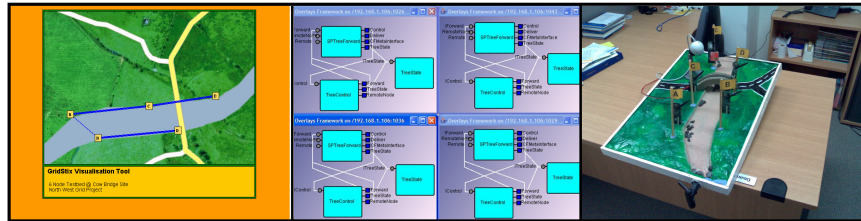
equipped with GPRS for off-site data dissemination. This computational capability comes at the expense of power consumption. During typical operation each Gumstix consumes up to 1W with a maximum power draw of up to 3W. This is not a significant problem however in our target domain, as it is feasible for these power requirements to be met using a high-capacity battery backed by solar panels.

As noted, GridStix are capable of maintaining ad-hoc on-site networking using either Bluetooth or 802.11b. This is useful as these networks have different properties and one can be selected over the other to reflect environmental conditions. For example, to detect the rate-of-flow of rivers, we employ cameras and a locally-executed image analysis algorithm. Individual nodes can detect coarse-grained changes in surface velocity; however, for more precise measurements, images need to be distributed to a number of nodes. To support this, we may switch network types from Bluetooth to 802.11b, which is better suited to distributing large data-sets such as images. The performance of the GridStix is discussed in more detail in [7].

More fine-grained adaptations are also performed at the overlay level, in that different overlays can be substituted for each other at run-time depending on conditions. For example, nodes are initially structured using a power-efficient Shortest-Path (SP) spanning tree. However, SP trees have low resilience. Therefore, when flooding is predicted we can increase resilience by dynamically substituting a more resilient Fewest-Hop (FH) tree. Thanks to GridKit's component-based nature, this can be achieved by simply replacing the 'forwarding' component.

## 2 Demonstration

Our demonstration will use a mixture of physical props, audience participation and visualisations to illustrate how a GridStix-based WSN, deployed to perform flood monitoring, can adapt its behaviour to best suit changing conditions. The demo installation (see Figure 1, right) comprises a three dimensional model of the actual site at which the WSN is deployed. Five Gumstix nodes are deployed on this map reflecting their real-world locations. The physical installation is accompanied by two visualizations: The first shows the current overlay network topology (see Figure 1, left), and the second shows the current software component configuration on each node (see Figure 1, middleware).



**Figure 1 – Demo Installation and Component Visualisation**

During the demonstration, the audience is invited to interact with the physical installation. This happens in two ways. First, participants are invited to simulate fast

flow conditions by rolling marbles down the model river bed (see Figure 1, right). The motion of the marbles is detected by a network camera and this causes the GridStix network to switch from Bluetooth to 802.11b, simulating the adaptation that occurs when an on-site camera detects increased flow rates. When the user stops rolling marbles, this simulates river flow returning to normal conditions and the WSN switches back to the Bluetooth to conserve power. These changes are reflected in the network visualization.

In the second mode of interaction, participants are invited to simulate increasing water depth. In a real-world deployment, depth predictions are made using a 'point-based' flood prediction algorithm executed on the local 'mini-grid'. Increased flood risk causes the WSN to adapt the spanning tree that is being used to disseminate data off-site. In particular, as outlined above, the configuration changes from a low-power shortest-hop spanning tree to a more reliable fewest-hop spanning tree by swapping the 'forwarding' component of the 'overlay' component framework running on each node (see Figure 1, middle). For the purposes of the demo, in order to simulate increasing water depth, users are invited to pour water onto the model river bed (see Figure 1, right). This action triggers a water sensor which causes the WSN to perform the above reconfiguration. This is reflected both in the component visualization, which shows the forwarding components of the spanning-tree overlay being swapped (see Figure 1, middle) and in the network visualization, which shows the rebuilding of the tree to conform to the fewest hop topology (see Figure 1, left). Following this adaptation sequence, users are invited to continue adding water, which, at a set level, simulates node failure due to immersion. This again is reflected in the network visualization as the node is removed from the network topology.

## 5. References

1. F. Pappenberger, K. Beven. et al., "Cascading model uncertainty from medium range weather forecasts (10 days) through a rainfall-runoff model to flood inundation predictions within the European Flood Forecasting System (EFFS)", *Hydrology and Earth System Science*, 9(4), pp381-393, 2005.
2. F. Pappenberger, P. Matgen and K. Beven, "The influence of rating curve and structural uncertainty on flood inundation predictions.", *Advances in Water Resources*. (in press) 2006.
3. Waysmall Computers "Gumstix Embedded Computing Platform Specifications", website: <http://gumstix.com/spexboards.html>.
4. G. Coulson, P. Grace, G. Blair et al, "Open Overlay Support for the Divergent Grid", in the proceedings of the UK E-Science All Hands Meeting, Nottingham, UK, September 2005.
5. G. Coulson, G. Blair et al "The Design of a Highly Configurable and Reconfigurable Middleware Platform", *ACM Distributed Comp Journal*, Vol. 15, No 2, pp109-126, April 2002.
6. Crossbow "Mica Motes and Sensors", website: <http://www.xbow.com/>
7. "An Intelligent and Adaptable Flood Monitoring and Warning System", Hughes D., Greenwood P., Coulson G., Blair G., Pappenberger F., Smith P., Beven K., to be published in the proceedings of the 5th UK E-Science All Hands Meeting (AHM'06), September 2006